Groundwater level modeling using multi-objective optimization hybridizing artificial intelligence

Fatemeh Barzegari Banadkooki (Barzegari@pnu.ac.ir)
Payame Noor University, Iran
Ali Torabi Haghighi
University of Oulu

Research Article

Keywords: Multi-objective genetic algorithm, Particle swarm optimization, Multi-layer perceptron, Groundwater, Arid region.

Posted Date: May 23rd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2915223/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Environmental Modeling & Assessment on November 3rd, 2023. See the published version at https://doi.org/10.1007/s10666-023-09938-6.

Abstract

Estimating groundwater level (GWL) fluctuations is essential for integrated water resource management in arid and semi-arid regions. This study promotes the multi-layer perceptron (MLP) learning process using hybrid evolutionary algorithms. This hybrid metaheuristic algorithm was applied to overcome MLP difficulties in the learning process, including its low conversions and local minimum. Also, the hybrid model benefits from the advantages of two objective function procedures in finding MLP parameters that result in a robust model regardless of over and under-estimating problems. These algorithms include none dominated sorting genetic algorithm (NSGA II) and multi-objective particle swarm optimisation (MOPSO) in different patterns, including MLP–NSGA-II, MLP–MOPSO, MLP-MOPSO–NSGA-II, and MLP-2NSGA-II–MOPSO. Temperature, precipitation and GWL datasets were used in various combinations and delays as model input candidates. Finally, the best model inputs were selected using the correlation coefficient (R^2). Input parameters include temperature and precipitation delays of 3, 6, and 9 months and GWL delays of 1 to 12 months. In the next step, the performance of the different combinations of MLP and hybrid evolutionary algorithms was evaluated using The root mean square error (RMSE), correlation coefficient (R), and mean absolute error (MAE) indices. The outcomes of these evaluations revealed that the MLP-2NSGA-II-MOPSO model, with an RMSE=0.073, R=0.98, and MAE=0.059, outperforms other models in estimating GWL fluctuations. The selected model benefits from the advantages of both MOPSO and NSGA-II regarding accuracy and speed. The results also indicated the superiority of multi-objective optimization algorithms in promoting MLP performance.

Introduction

GWL (Groundwater level) prediction is a complicated process, and its modeling is incredibly challenging (Adnan et al., 2022) because of its nature and dependence on several hydrogeological and geological interactions (Agoubi & Kharroubi, 2019). For example, the heterogeneity of the subsurface environment, data limitation, nonlinear behavior, and model complexity through the coupling of different equations are crucial challenges for GWL modeling. On the other hand, GWL modeling is an essential tool for groundwater management. Accurate GWL prediction can provide valuable information for water allocation, contaminant transport, climate change adaptation and mitigation plan, urban and rural planning, and assess the impact of overexploitation of GW. Therefore due to the growing global water scarcity and accelerating global warming, there is a growing need for accurate groundwater resource quantification for better management along with environmental protection (Li et al., 2015; Noori et al., 2021; Torabi Haghighi et al., 2020). Furthermore, precise GWL modeling can optimize the monitoring of GWL, which has time and resource limitations (Kazemi et al., 2021) and needs significant in-situ measurements (Lee et al., 2019) and interpretation of data quality.

The GWL fluctuations can be due to (i) climate change and variability and (ii) anthropogenic activity, i.e., GW exploitation (Malekinezhad & Banadkooki, 2018). Therefore, GWL modelling needs deep analysis, extensive and reliable data sources, and an understanding of interactions between different environmental parameters (Liang & Zhang, 2015). In recent decades many attempts have been made to develop accurate

and robust models to predict GWL fluctuations properly. These attempts resulted in a variety of GWL models that can be categorized into four groups, (i) the univariate time series models (Khorasani et al., 2016; Retike et al., 2022), (ii) conceptual models (Hong et al., 2016; Lyazidi et al., 2020), (iii) the physically-based models (Bailey et al., 2020; Condon et al., 2021) and (iv) the artificial intelligence (AI) models (Adnan et al., 2019; Besaw et al., 2010; Chang et al., 2015; Ghazi et al., 2021; Gholami et al., 2015; Ghose et al., 2018; Guzman et al., 2017; Guzman et al., 2019; Lallahem et al., 2005; Lee et al., 2019; Malik & Bhagwat, 2021; Mohanty et al., 2015; Nourani et al., 2015; Nourani & Mousavi, 2016; Peng et al., 2017; Ranjithan, 1993; Roshni et al., 2020; Sahoo et al., 2017; Sun et al., 2016; Zhang et al., 2019; Zhou et al., 2017). The physically-based and conceptual models have faced some limitations, such as low prediction accuracy (36) and the need for many input parameters (Rajaee et al., 2019). The univariate time series models also have the challenge of weak performances in modeling nonlinear and non-stationary parameters such as climate data series (Moghaddam et al., 2019).

However, in the last two decades, many studies have investigated the applicability of AI models in GWL modeling, e.g., ANNs (Rogers), support vector machines (SVM) (Zhou et al., 2017), and fuzzy-based models (Nadiri et al., 2019). Due to the limitation of the single AI models (Chang et al., 2015; Ghorbani et al., 2017; Kashiwao et al., 2017; Kombo et al., 2020; Natarajan & Sudheer, 2019; Nourani et al., 2014; Pham et al., 2019; Yaseen et al., 2019; Zhang, 2003), Most recently, the application of hybrid AI models was considered in GWL simulation and modelling processes. These hybrid models include a large variety of AI models such as SVM, ANN, neuro-fuzzy inference system (ANFIS) (Jang, 1993), emotional ANN (EANN) (Roshni et al., 2020), multilayer perceptron (MLP) (Afan et al., 2014; Deo & Şahin, 2015; Rakhshandehroo et al., 2012), generalized regression neural network (GRNN) (Roshni el al. 2020) and backpropagation neural network (BNN) (Maiti & Tiwari, 2013). These novel models are usually used with optimization algorithms (e.g., PSO, GA, whale algorithm (WA), weed algorithm (WA), differential evolution (DE), cat swarm optimization (CSO), and quantum-behaved particle swarm optimisation (QPSO) to enhance modelling and forecasting. Table 1 provides some recent studies that used hybrid AI models in GWL modelling.

Table 1- **The list of the published papers in the literature on the hybridized ANN models to GWL modeling**.

No.	Author (year)	Hybrid AI models	Input parameters	Frequency	Performance metrics	Area
1	Dash et al. (2010) (Dash et al., 2010)	ANN-GA	GWL	Monthly	R, E, MAE, IOA, RMSE	Mahanadi river basin, India
2	Jalalkamali and Jalalkamali, (2011) (Jalalkamali & Jalalkamali, 2011)	GA-ANN, FFNN, RNN	Piezometers, T, R	Monthly	R^2, RMSE, MAPE	Kerman plain (Kerman, Iran)
3	Sudhee et al. (2012) (Sudheer & Shashi, 2012)	SVM-QPSO, ANN	GWL	Monthly	EFF, R, RMSE	Andhra Pradesh, India
4	Jha and Sahoo, (2015) (Jha & Sahoo, 2015)	ANN-GA	GWL, R, T	Monthly	R^2, RMSE, IOA, NSE, Bias, CV	Konan basin, Kochi, India
5	Yang et al. (2015) (Yang et al., 2015)	WA-ANN, ANN	GWL	Monthly	RMSE, R, EFF	Fujian, China
6	Nourani et al. (2015), (Nourani et al., 2015)	WA-ANN, ANN	GWL, R,	Monthly	R^2, RMSE	Ardabil plain, Iran
7	Huang et al. (2017) (Huang et al., 2017)	PSO-SVM, PSO-BPNN	GWL	Daily	RMSE, NSE, R^2	Gorges Reservoir Area, China
8	Balavalikar et al. (2018) (Balavalikar et al., 2018)	PSO-ANN, ANN	GWL	Monthly	R^2, RMSE, R, MAE, MAPE	Brahmavar, Kundapur and Hebri In Udupi district, India
9	Supreetha et al. (2019) (Supreetha et al., 2019)	PSO-ANN, ABC-ANN	GWL, P	Monthly	RMSE, MAE, MAPE	Karnataka, India
10	Roshni, (2020) (Roshni et al., 2020)	EANN-GA, EANN, GRNN, FFNN	P, GWL	Monthly	NSE, RMSE,Bias	Konan groundwater basin, Japan
11	Banadkooki et al. (2020) (Banadkooki et al., 2020)	RBF-WOA, MLPWOA	R, T, GWL	Monthly	NSE, MAE	Yazd, Iran

12	Adnan et al., (2023) (Adnan et al., 2022)	ELM-JFO, ELM-WOA, ELM-HHO	GWL	Monthly	RMSE, MAE, R^2	Bangladesh
13	Pandey et al., (2020) (Pandey et al., 2020)	GA-ANN	GWL	Seasonal	R2, RMSE, MAD, CE, APE, PI	INDIA
14	Cui et al., (2022) (Cui et al., 2022)	ANFIS-PSO, ANFIS-IA-GWO, ANN-PSO, ANN-PSO-GWO	GWL	Monthly	NRMGESE, K	Bangladesh
15	Ehteram et al., (2022) (Ehteram et al., 2022)	ANN-PSO, ANN-GA, ANN-RSA, ANN-SSA	GWL	Monthly	RMSE	Yazd, Iran
16	Zhang et al., (2022) (Zhang et al., 2022)	GA-BP, BP	GWL	Monthly	RMSE, MAPE, NSE	China

According to previous work, the hybridization of GA or PSO with ANN models has shown distinguished improvement in GWL modeling (Balavalikar et al., 2018; Cui et al., 2022; Dash et al., 2010; Ehteram et al., 2022; Huang et al., 2017; Jalalkamali & Jalalkamali, 2011; Jha & Sahoo, 2015; Supreetha et al., 2019). As two well-known optimization algorithms, GA and PSO have deniable effects on improving the performance accuracy or optimum outputs of the ANN models. GA can support ANN in reducing the local minimum, which is one of the main limitations of ANN (Kisi & Shiri, 2012; Tahmasebi & Hezarkhani, 2009; Yadav et al., 2018). GA can improve the topology of ANN and its learning parameters and is equipped with appropriate local search techniques for locating optimal solutions to non-linear problems (Adib & Mahmoodi, 2017; Tahmasebi & Hezarkhani, 2009). However, PSO is more efficient than GA regarding speed and memory requirements, while it has lower accuracy and practicality. Therefore the combination of PSO and GA can decrease the drawbacks of PSO in solving optimization problems. Although some studies have combined GA or PSO algorithms with ANN to model GWL, to our knowledge, hybrid PSO and GA models with ANN have not been considered yet.

Furthermore, previous studies selected ANN parameters by solving single-objective problems to minimize mean square error (MSE) (Balavalikar et al., 2018; Dash et al., 2010; Ehteram et al., 2022; Jha & Sahoo, 2015; Pandey et al., 2020; Roshni et al., 2020; Zhang et al., 2022). There are a few drawbacks; MSE is a

non-linear combination of mean error (bias) and variance, and minimizing the MSE does not imply minimizing both components (variance and bias). Attempting to minimize variance may result in an under-fitted model, whereas attempting to minimize bias may result in an over-fitted model. By minimizing the two parameters, a robust model can be developed (bias and variance). In this research, we planned to promote ANN structure using the advantages of combined GA and PSO algorithms to predict groundwater fluctuations in the Yazd–Ardakan aquifer in central Iran from 2000 to 2014. In addition, all parameters of the ANN model were optimally selected using two objective optimizations (NSGA-II and MOPSO).

Materials and Methods

In this part, we first introduce our case study and then present the methodology for predicting GWL in the Yazd-Ardakan aquifer using a hybrid NSGA-II and MOPSO algorithm with an MLP model. NSGA-II-MOPSO combines the strengths of two powerful optimization methods: Non-dominated Sorting Genetic Algorithm II (NSGA-II) and Multi-Objective Particle Swarm Optimization (MOPSO). To address multi-objective optimization problems, this hybrid approach was developed to overcome the limitations of traditional single-objective optimization techniques. The main objective of MLP-2NSGA-II-MOPSO is to maximize the number of objectives achieved in a given problem. The algorithm begins by defining a set of possible solutions that are then evaluated based on a set of objective functions. In the next step, these solutions are sorted using NSGA-II and refined using MOPSO. The MOPSO algorithm uses a local search approach to refine the solutions. As a final step, we will select the best solutions based on the trade-offs between the objectives. As a result, MLP-2NSGA-II-MOPSO is a powerful and efficient optimization technique that can be applied to a wide range of multi-objective optimization problems. The combination of NSGA-II and MOPSO makes it a powerful tool for addressing complex optimization problems that require multiple trade-offs.

Study area

The Yazd-Ardakan aquifer is the region's primary water source and is essential for industrial and mineral development. The study area is characterized by a hot and arid climate and is located in Iran's Yazd province at 53° 30–55° east longitude and 31° 15–32° 30 north latitude (Fig. 1-a). Mean annual precipitation and temperature are 102.3 mm and 19.1°C, respectively (Fig. 2). The limitation of water resources, accelerated area development, and increased competition among exploiters resulted in a significant decline in aquifer storage (Fig. 1-b).

The minimum and maximum GWL of the Yazd-Ardakan aquifer were observed in October and April. Figure 2 illustrates the average monthly minimum and maximum temperatures and precipitation values.

To predict groundwater fluctuations in the Yazd–Ardakan aquifer, we applied climate and GW data from 2000 to 2014. Table 2 summarizes some of the model's input variables. The model structure was created by dividing data into 80% and 20% for training and testing. Input variables for the model are monthly precipitation, temperature, and GWL with varying monthly delays (1–12 months). The correlation

coefficient between variables and groundwater level was calculated to determine whether input parameters experienced significant delays.

Table 2
Characteristics of model input variables of Yazd-Ardakan aquifer.

Parameter	Average Temperature	Average Rainfall	Water level
Train			
Min	8.1	3.2	1032.28
Max	31.4	15.1	1035.987
Average	19.5	5.7	1034.15
Variation coefficient	0.018	0.18	1.14
Test			
Min	6.9	5.8	1031.84
Max	31.9	13.8	1036.53
Average	19.3	4.8	1034.17
Variation coefficient	0.045	0.16	1.24

MSE (Mean Squared Error) is an important metric in machine learning. Optimizing the MSE does not guarantee optimization of both mean error (bias) and variance. As a result, models can be overfitted or under fitted. It is important to minimize bias and variance to ensure a robust model. The bias-variance trade-off states that reducing bias increases variance and vice versa. In order to achieve a good model, both components must be balanced. Thus, bias-variance trade-offs need to be considered when optimizing MSEs. To achieve this goal, two objective functions are considered. In order to achieve our objective, we must minimize bias and variance.

Multi-Layer Perceptron (MLP)

The Yazd-Ardakan aquifer's GWL fluctuations were modeled using MLP with various input parameters. Rainfall, temperature with delays of 3, 6, and 9 months, and groundwater table with various delays, ranging from 1 to 11 months, were chosen as input parameters. MLPs are a prevalent artificial neural network that models non-linear relationships between input variables (Park & Lek, 2016). MLP has been successfully used to model hydrological processes such as sediment transport (Aksoy & Mohammadi, 2016; Rajaee et al., 2009), time series modelling (Kişi, 2010), and groundwater simulations (Gholami et al., 2015; Rajaee et al., 2019; Sun et al., 2016). The model structure comprises three distinct types of layers (Fig. 3).

1) The input layer contains the input parameters and communicates critical information to the subsequent layer. The tan-sigmoid approach was used throughout the modeling process as a transfer function.

2) A hidden layer tasked with the responsibility of computing processes and learning algorithms. The Levenberg-Marquardt algorithm was considered a learning algorithm in this study.

3) Output layer. This layer consists of results extracted from previous layers.

Recent research indicates that a single hidden layer is sufficient to solve non-linear functions and reduce the complexity of the structure of the ANN model (Hornik et al., 1989; Tang et al., 1991). Thus, a single hidden layer network was used in this study to predict GWL fluctuations. The Levenberg-Marquardt algorithm developed the ANN model due to its computational efficiency and suitability for training (Hagan & Menhaj, 1994). All parameters of the ANN model were optimally selected in this research using two objective optimizations.

2) Non-dominated sorting based multi-objective evolutionary algorithm (NSGA-II)

NSGA-II is an improved version of Deb's non-dominated sorting genetic algorithm from 2000. This well-known evolutionary algorithm has been the subject of extensive research. It is constructed using a genetic algorithm with a multi-objective process. A key advantage of NSGA-II is its ability to generate solutions representing the entire Pareto optimal front. Hence, it can provide a range of better solutions than a single one. A major advantage of NSGA-II is its ability to handle multiple objectives. It can quickly and efficiently identify the optimal solution for a given set of objectives. Another advantage of NSGA-II is that it can be scalable. It can be easily scaled up to handle complex optimization problems with many variables. Consequently, it is ideal for large-scale optimizations.

Survival of the fittest is the main objective of the GA algorithm. This procedure enables them to transfer genomes from one generation to the next. This means that improved adaptations are formed in subsequent generations. The NSGA-II algorithm was used to optimize the selection of ANN model parameters. The process begins with an initial population (chromosomes) and evaluates the fitness values of all chromosomes using both objective functions (bias and variance). The following non-dominated fronts were calculated using non-domination rules: obj.1 (i) > obj.1 (j) and obj.2 (i) > = obj.2 (j) or obj.1 (i) > = obj.1 (j) and obj.2 (i) > obj.2 (j), where i and j are chromosome numbers. Subsequently, various genetic operations, such as selection crossover and mutation, were used to improve the sorted initial chromosomes concerning two objective functions. The initial chromosome count was set to 50 to balance diversity and computational speed. Gaussian mutation and double point crossover were used to solve the optimization problem in this study (Fig. 4).

3) Particle Swarm Optimization

The PSO is considered a bio-inspired method for optimizing non-linear problems (Kennedy, 1997). The behavior of biological populations inspires this method to simulate the navigation and foraging behavior

of a flock of birds or a school of fish. PSO is also known for its efficiency. Compared to other algorithms, it explores the search space faster and converges faster. Additionally, it is computationally inexpensive and easily parallelizable, making it ideal for large-scale optimization.

The PSO algorithm employs a population (labeled swarm) source of candidate solutions (labeled particles). The particle combinations are moved around the search space using simple formulas. Two fundamental rules govern particle movement: their own best optimization found in the search space and the best-known position of any particle.

PSO algorithm includes three steps: generation of particle positions and speeds, updating particle speeds, and finally, updating particle positions. Each particle was repositioned in the space in response to the speed updates. At the start of the optimization process, each initial particle's position and speed are randomly defined. The PSO algorithm proceeds through the stages below until the end condition is satisfied.

1- Estimating the objective value for all particles.

2- Updating W, C_1, and C_2 values mentioned in Eq. 2.

3- Updating the personal best and the global best.

4- Estimating each particle's velocity vector using Eq. 1.

5- Calculating the subsequent location of each particle using Eq. 2.

Each optimization stage updates the position of particles using Eq. 1.

$$\overrightarrow{X_i} = \overrightarrow{X_{i(t)}} + \overrightarrow{V_{i(t+1)}}$$

1

X_i indicates the i_{th} particle's position at the t_{th} iteration in this equation. Eq. 2 is used to calculate the velocity vector.

$$\overrightarrow{V_{i(t+1)}} = w\overrightarrow{V_{i(t)}} + c_1 r_1 \left(\overrightarrow{P_{i(t)}} - \overrightarrow{X_{i(t)}}\right) + c_2 r_2 (\overrightarrow{G_{(t)}} - \overrightarrow{X_{i(t)}})$$

2

Where X_i is the i_{th} particle's position in the t_{th} iteration, V_i denotes the velocity of the i_{th} particle at the t_{th} iteration, w denotes inertial weight, $c_{1,}$ and c_2 denote individual and social coefficients, respectively, and r_1 and r_2 are random numbers between 0 and 1, respectively. P_i denotes the best position of the first particle until the i_{th} iteration, and G(t) denotes the best position obtained by the entire swarm up to the i_{th} iteration. The PSO is capable of pursuing multiple objectives (Coello et al., 2004). One strategy is to maximize a new

objective function for operating multiple objectives that are formulated using conventional numerical optimization. by multiplying the objectives (O_i, i = 1,2...,n) by their corresponding user-defined positive weights (w_i, i = 1,2...,n that $\sum_{i=1}^{n} w_i = 1$), which results in $\sum_{i=1}^{n} w_i O_I = 1$. These weights can be fixed or changed dynamically throughout the optimization process.

A genetic algorithm and particle swarm optimization combined with a multi-layer perceptron neural network could be used to improve the topology of MLP and its variables. Thus, in this study, various NSGA-II and MOPSO models were used to determine the number of hidden neurons, weights, and bias values for an MLP model. The hybrid MLP-MOPSO model's flowchart is shown in Fig. 5, and the MLP-MOPSO-NSGA-II and MLP-2NSGA-II-MOPSO models' schematic diagrams are shown in Fig. 6. As a robust algorithm, MOPSO handles a wide range of problems with varying complexities and sizes. This algorithm can handle problems with a wide range of parameters and objectives, making it suitable for many applications. The MOPSO is created based on the following levels:

1. Define the problem: The problem needs to be formulated as a multi-objective optimization problem with more than one objective function that needs to be optimized simultaneously.
2. Initialization: Initialize the population of particles with random values within predefined boundaries.
3. Evaluate fitness: Calculate the fitness of each particle based on the objective functions.
4. Update particle position: Update the position of each particle based on its velocity and the position of the best particle it has experienced so far.
5. Eliminate invalid solutions: Discard any particle that violates the problem constraints.
6. Pareto dominance: Use Pareto dominance to compare the particles in the current population and select the non-dominated particles.
7. Update personal best: Each particle stores its best position so far.
8. Update global best: The algorithm selects and stores the best position.
9. Update velocity: Update the particle velocity based on the best particle in the population.
10. Termination: When the algorithm reaches its maximum number of iterations or reaches the desired level of accuracy, it terminates.

Table 3 illustrates various combinations of MOPSO and NSGA-II algorithms with MLP models. The training data set includes rainfall, temperature, and piezometric data from the study aquifer with significant delays. 104 records were used to model each variable, and all models were trained using the same data.

Table 3
MLP model details and parameters

Details	MLP	ML -GA	MLP-MOPSO	MLP-NSGA-II	MLP-MOPSO-NSGA-II	MLP-2NSGA-II-MOPSO
Type of ANN model	MLP	MLP	MLP	MLP	MLP	MLP
Training algorithm	-	GA	MOPSO	NSGA-II	50%MOPSO & 50% NSGA-II	66% (NSGA-II) & 33% (MOPSO)
Activation function	tan-sigmoid	tan-sigmoid	tan-sigmoid	tan-sigmoid	tan-sigmoid	tan-sigmoid
Number of the input layer	1	1	1	1	1	1
Number of hidden layers	1	1	1	1	1	1
Number of hidden neurons	5	5	5	5	5	5
C1	1.5	1.5	1.5	1.5	1.5	1.5
C2	1.5	1.5	1.5	1.5	1.5	1.5
Number of generations	-	100	-	100	100	100
Particle population	-	-	100	-	100	100
Function for performance	MSE	MSE	MSE	MSE	MSE	MSE

The GA variables, including selection method, population size, crossover rate, mutation rate, and generation number, were determined using the hit and trial procedure. Table 4 summarizes the NSGA-II parameters.

Table 4
NSGA-II parameter setting

Parameters	Setting
Population size	50
Selection rate	80%
Crossover rate	70%
Mutation rate	30%
Crossover type	Two-points
Maximum number of iterations	200
Basis of chromosome selection	Bias- Variance
Stopping criteria	0.001m^2

We evaluated the models using RMSE, MAE (Willmott & Matsuura, 2005), and R (Galton & British, 1885) indices obtained from Eqs. 3–5 for the training and testing datasets, respectively.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left((O_i - P_i) \right)^2}{n}}$$

3

$$R = \frac{\sum_{i=1}^{n} \left(O_i - \bar{O_i} \right) \left(P_i - \bar{P_i} \right)}{\sum_{i=1}^{n} \left(O_i - \bar{O_i} \right) \sum_{i=1}^{n} \left(P_i - \bar{P_i} \right)}$$

4

$$MAE = \frac{\sum_{i=1}^{n} |(O_i - P_i)|}{n}$$

5

where O_i, $\bar{O_i}$, p_i and $\bar{p_i}$ denote the observed data, the average of the observed data, the predicted value, and the average of the predicted value, respectively. Also, the n value refers to the number of samples.

Uncertainty of model parameters

ANN models face uncertainty with and without neural algorithms. The parameters of model structures in the neural network can be uncertain. They can include optimized weight and bias values, which have uncertainty despite being calculable by the obtained algorithms. Generalized likelihood uncertainty estimation (GLUE) is a popular method to measure uncertainty in the hydrological analysis. It is extensively used to calculate uncertainty in various areas: for instance, in river flow and flood simulations, dust estimation, and hydrological modeling. This study uses the GLUE method to estimate the uncertainty of neural network models due to model parameters:

- The prior probability distribution of each parameter estimated should be quantified for which normal uniform distribution is typically used.
- The number of samples is provided from each model parameter based on the Monte Carlo method and in line with the previous distribution. The study samples were 5,000.
- The models are rerun using model parameters to obtain an estimated output.
- The correct likelihood values of each parameter are calculated. The GLUE method uses the probability likelihood function by considering an objective function. The objective function mainly consists of Nash–Sutcliffe values (Sheng et al., 2019) (Eq. 6).

$$P\left(Y|\theta_i\right) = 1 - \frac{\sum_{n=1}^{N}\left[Y_O - \hat{Y}_O\left(\theta_i\right)\right]^2}{\sum_{n=1}^{N}\left(Y_O - \bar{Y}_O\right)^2}$$

6

Where, $P\left(Y|\theta_i\right)$: likelihood value, Y_O: Observed data, N: number of data, $\bar{Y}$: Average observed value, and $\hat{Y}_O\left(\theta_i\right)$: estimated value.

In the next step, a threshold is defined: parameter values are rejected if the likelihood value does not exceed this threshold. Finally, the posterior probability distribution is determined using Eq. (7):

$$P\left(\theta_i|Y\right) = \frac{P\left(\theta_i\right) * P\left(Y|\theta_i\right)}{\sum_{i=1}^{N} P\left(Y|\theta_i\right)}$$

7

where, $P\left(\theta_i|Y\right)$: posterior distribution, $P\left(Y|\theta_i\right)$: likelihood value, and $P\left(\theta_i\right)$: prior distribution,

Two indicators measure the uncertainty values of the models. The first indicator is represented by p (Eq. 8): higher values indicate that a higher percentage of points falls within the 95% confidence band, thus less uncertainty. The second indicator is represented by r (Eq. 9): the higher its values, the higher the uncertainty and confidence bandwidth.

$$p = \frac{NGW}{n} * 100$$

8

$$r = \frac{\sum_{t=1}^{n} (GW_U - GW_L)}{n\sigma_o}$$

9

Where, GW_U: upper bound of groundwater level, GW_L: lower bound of groundwater level, σ_o: standard deviation, NGW: the number of observations enveloped by 95% prediction uncertainty band, n: number of data

Results and Discussion

The correlation between different input parameters (lagged temperature, precipitation, and GWL) with GWL for modeling is presented in Table 5.

Table 5
The correlation between precipitation, temperature, and groundwater data and groundwater levels

Temperature	r	Precipitation	r	GW level	r
T(t-1)	-0.2	R(t-1)	0.25	G(t-1)	0.99
T(t-2)	-0.34	R(t-2)	0.44	G(t-2)	0.99
T(t-3)	**-0.82**	R(t-3)	**0.91**	G(t-3)	0.98
T(t-4)	0.65	R(t-4)	0.76	G(t-4)	0.98
T(t-5)	-0.51	R(t-5)	0.65	G(t-5)	0.98
T(t-6)	**-0.78**	R(t-6)	**0.83**	G(t-6)	0.98
T(t-7)	-0.45	R(t-7)	0.51	G(t-7)	0.97
T(t-8)	-0.62	R(t-8)	0.45	G(t-8)	0.98
T(t-9)	**-0.75**	R(t-9)	**0.86**	G(t-9)	0.98
T(t-10)	-0.56	R(t-10)	0.35	G(t-10)	0.99
T(t-11)	-0.32	R(t-11)	0.2	G(t-11)	0.99
T(t-12)	-0.42	R(t-12)	0.24	G(t-12)	0.99

As seen the Table 5 that precipitation and temperature with 3, 6, and 9 months lags have more influence on GWL than others and are selected as inputs for GWL modeling. The correlation between GWL and GWL different lags range between 0.97 to 0.98, thus, G (t-1), G (t-1), …. G (t-12) were selected as input parameters for GWL modeling.

Table 6 sums up the outcomes of the testing and training stages of the hybrid and stand-alone MLP models in GWL modeling.

Table 6
Performance evaluation of MLP models

Model/criteria	**R**		**RMSE**		**MAE**	
	Test	Train	Test	Train	Test	Train
MLP	0.86	0.899	0.181	0.17	0.151	0.21
MLP-GA	0.89	0.912	0.178	0.158	0.156	0.14
MLP-NSGA-II	0.9576	0.9606	0.115	0.1116	0.092	0.088
MLP-MOPSO	0.9034	0.9225	0.123	0.12151	0.148	0.13
MLP-2NSGA-II-MOPSO	**0.986**	**0.972**	**0.073**	**0.092**	**0.059**	**0.069**
MLP-MOPSO-NSGAII	0.963	0.960	0.0972	0.111	0.102	0.095

Of different models, MLP-2NSGA-II-MOPSO outperforms other models with the highest R (0.972) and the lowest RMSE (0.092) and MAE (0.069) in GWL estimation (Table 6). As seen, combining the optimization algorithm with the MLP improves the model performance and forecasting accuracy (hybridization of MLP with two-objective algorithms (NSGA-II)). Previous studies also confirmed hybrid ANN models with two-objective optimization algorithms perform better than single-objective optimization (Trưởng & Dao, 2020; Yadav et al., 2021).

MLP-NSGA-II model provides better improvement; decreases in RMSE and MAE of the MLP-GA model from 0.18 to 0.11 and from 0.15 to 0.095 and increases in R from 0.921 to 0.96 in the training stage. The train and test stages of the stand-alone MLP model and the proposed hybrid models MLP-NSGA-II, MLP-MOPSO-, MLP-2NSGA-II-MOPSO-, and MLP-MOPSO-NSGA-II-, are depicted in Fig. 7.

The results show that hybridization of the MLP with optimization algorithms improves the model accuracy and performance compared to stand-alone MLP (Fig. 7). Also, two-objective algorithms, including NSGA-II and MOPSO, are used, have better performance than single-objective algorithms such as GA. The best

performance is related to MLP-2NSGA-II-MOPSO model. The model accuracy was improved in previous studies such as (32, 58, 61, 64, 67, 69, and 70).

The hybrid MLP-2NSGA-II-MOPSO model benefits from the integration of the search engines of both algorithms to generate a new population. Therefore it can establish the divers and wast pareto optimal solution. During each iteration, according to the ranking, the population is separated into halfs, the analysis is performed by NSGA-II, using upper half of the population. MOPSO algorithm modify the lower half effectively using different operators.

MOPSO-NSGAII is a multi-objective optimization algorithm that combines the strengths of both Multi-Objective Particle Swarm Optimization (MOPSO) and the Non-Dominated Sorting Genetic Algorithm II (NSGAII). This algorithm has proven to be more effective for multi-objective optimization problems than NSGAII. MOPSO-NSGAII maintains a larger population size and has a better convergence rate than NSGAII. With a larger population size, the algorithm can explore a broader search space and identify more diverse solutions. Hence, more non-dominated solutions can be identified, and convergence is faster. To store non-dominated solutions, MOPSO-NSGAII uses a dynamic archive. By comparing the current population with the archive, the algorithm can detect new non-dominated solutions. A non-dominated solution set can be reached more quickly. In addition, MOPSO-NSGAII employs a local search strategy that further improves accuracy. The algorithm finds the best solutions more accurately by performing local searches. MOPSO-NSGAII outperforms NSGAII through a larger population, dynamic archives, and local search operators. The combination of these features makes MOPSO-NSGAII more effective in multi-objective optimization.

Uncertainty of model parameters

Results indicated that using two-objective algorithms has effectively improved the model performance (Fig. 8). For example, NSGA-II improved P-value from 0.89 to 0.91 and R from 0.24 to 0.19 compared to the MLP-GA model. Also, the MOPSO-MLP model in the same process affects the model performance, increasing the P-value from 0.89 to 0.9 and decreasing R from 0.24 to 0.2. Also, results indicated that the MLP-2NSGA-II-MOPSO model has lower uncertainty than other models due to having the highest values of P and lowest values of R. It means that a combination of the two algorithms (NSGA-II and MOPSO) improves the model accuracy by using more search agents and operators. Further, the MLP model has the highest uncertainty due to the lowest P-value and the highest R-value. NSGA-II-MOPSO is a meta-heuristic method that can be used to optimize the parameters of the MLP model. It uses an evolutionary algorithm to search for the optimal parameters, taking into account a number of objectives simultaneously. Thereby it can reduce the uncertainty of the MLP model since it can find the optimal settings for the model.

In general, the uncertainty of the model outputs is caused by three primary sources: the first source is the uncertainty of the data and knowledge, the second one, called parametric uncertainty, is related to unknown model parameters; and the third one, known as the structural uncertainty, is derived from physical complexity of the phenomena. In this work, the main source of the uncertainty is the parametric

one derived from the hybrid models of GWL prediction due to the regulative parameters and weights created in the training stage of models and the lack of historical data on groundwater withdrawal.

The main advantage of NSGA-II is that it avoids overfitting, which is one of the main problems of MLP models. Overfitting occurs when the model is too complex, which results in poor generalization and accuracy. The NSGA-II prevents this problem by identifying the best solution that balances complexity and accuracy. The NSGA-II technique effectively prevents MLP overfitting. The algorithm uses evolutionary algorithms to select the best solution from a set of potential solutions, considering multiple objectives such as accuracy, generalization, and overfitting. As a result, the optimal solution can be selected with an optimal balance between accuracy and complexity.

Conclusion

This paper simulated GWL fluctuations of the Yazd-Ardakan aquifer in central Iran using MLP hybridized with various evolutionary algorithms. Evolutionary algorithms, including NSGA-II and MOPSO, were hybridized with MLP in different patterns such as MLP-NSGA-II-, MLP-MOPSO-, MLP-MOPSO–NSGA-II and MLP-2NSGA-II–MOPSO. Using two-objective optimization algorithms will result in an optimal selection of ANN's parameters. The RMSE, R, and MAE indices assessed the model's performance. Temperature and precipitation delays of 3, 6, and 9 months and groundwater level delays of 1 to 12 months were significant at the 5% level as input parameters. The results indicate that the MLP-2NSGA-II-MOPSO- model, with an RMSE = 0.073, R = 0.98, and MAE = 0.059, outperforms other models in estimating GWL fluctuations. Uncertainty analysis showed the same results and verified the best operation of the MLP-2NSGA-II-MOPSO model. Also, the primary source of the uncertainty analysis in this paper is the parametric one derived from the hybrid models of GWL prediction due to the regulative parameters and weights created in the training stage of models and the lack of historical data on groundwater withdrawal. The second one is more important in this study.

Declarations

Ethical approval This article does not contain any experiments with human participants or animals, performed by any of the authors.

Competting interests The authors have no relevant financial or non-financial interests to disclose.

Author Contributions Data collection, algorithm programming and writing: FBB. Programming, writing, review and editing of the paper: ATH. All authors read and approved the final manuscript.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Availability of data and materials All data generated in this study are available in this paper.

Informed Consent All authors approved the final manuscript.

References

1. Adib, A., & Mahmoodi, A. (2017). Prediction of suspended sediment load using ANN GA conjunction model with Markov chain approach at flood conditions. *KSCE Journal of Civil Engineering*, *21*, 447-457.
2. Adnan, R. M., Dai, H.-L., Mostafa, R. R., Islam, A. R. M. T., Kisi, O., Heddam, S., & Zounemat-Kermani, M. (2022). Modelling groundwater level fluctuations by ELM merged advanced metaheuristic algorithms using hydroclimatic data. *Geocarto International*, *38*(1). https://doi.org/10.1080/10106049.2022.2158951
3. Adnan, R. M., Malik, A., Kumar, A., Parmar, K. S., & Kisi, O. (2019). Pan evaporation modeling by three different neuro-fuzzy intelligent systems using climatic inputs. *Arabian Journal of Geosciences*, *12*, 1-14.
4. Afan, H. A., El-Shafie, A., Yaseen, Z. M., Hameed, M. M., Wan Mohtar, W. H. M., & Hussain, A. (2014). ANN Based Sediment Prediction Model Utilizing Different Input Scenarios. *Water Resources Management*, *29*(4), 1231-1245. https://doi.org/10.1007/s11269-014-0870-1
5. Agoubi, B., & Kharroubi, A. (2019). Groundwater depth monitoring and short-term prediction: applied to El Hamma aquifer system, southeastern Tunisia. *Arabian Journal of Geosciences*, *12*, 1-10.
6. Aksoy, H., & Mohammadi, M. (2016). Artificial neural network and regression models for flow velocity at sediment incipient deposition. *Journal of Hydrology*, *541*, 1420-1429.
7. Bailey, R. T., Bieger, K., Arnold, J. G., & Bosch, D. D. (2020). A new physically-based spatially-distributed groundwater flow module for SWAT+. *Hydrology*, *7*(4), 75.
8. Balavalikar, S., Nayak, P., Shenoy, N., & Nayak, K. (2018). Particle swarm optimization based artificial neural network model for forecasting groundwater level in Udupi district. AIP Conference Proceedings,
9. Banadkooki, F. B., Ehteram, M., Ahmed, A. N., Teo, F. Y., Fai, C. M., Afan, H. A., Sapitang, M., & El-Shafie, A. (2020). Enhancement of groundwater-level prediction using an integrated machine learning model optimized by whale algorithm. *Natural resources research*, *29*, 3233-3252.
10. Besaw, L. E., Rizzo, D. M., Bierman, P. R., & Hackett, W. R. (2010). Advances in ungauged streamflow prediction using artificial neural networks. *Journal of Hydrology*, *386*(1-4), 27-37.
11. Chang, J., Wang, G., & Mao, T. (2015). Simulation and prediction of suprapermafrost groundwater level variation in response to climate change using a neural network model. *Journal of Hydrology*, *529*, 1211-1220.
12. Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on evolutionary computation*, *8*(3), 256-279.
13. Condon, L. E., Kollet, S., Bierkens, M. F., Fogg, G. E., Maxwell, R. M., Hill, M. C., Fransen, H. J. H., Verhoef, A., Van Loon, A. F., & Sulis, M. (2021). Global groundwater modeling and monitoring: Opportunities and challenges. *Water Resources Research*, *57*(12), e2020WR029500.
14. Cui, F., Al-Sudani, Z. A., Hassan, G. S., Afan, H. A., Ahammed, S. J., & Yaseen, Z. M. (2022). Boosted artificial intelligence model using improved alpha-guided grey wolf optimizer for groundwater level prediction: Comparative study and insight for federated learning technology. *Journal of Hydrology*, *606*. https://doi.org/10.1016/j.jhydrol.2021.127384

15. Dash, N. B., Panda, S. N., Remesan, R., & Sahoo, N. (2010). Hybrid neural modeling for groundwater level prediction. *Neural Computing and Applications*, *19*, 1251-1263.
16. Deo, R. C., & Şahin, M. (2015). Application of the Artificial Neural Network model for prediction of monthly Standardized Precipitation and Evapotranspiration Index using hydrometeorological parameters and climate indices in eastern Australia. *Atmospheric Research*, *161-162*, 65-81. https://doi.org/10.1016/j.atmosres.2015.03.018
17. Ehteram, M., Kalantari, Z., Ferreira, C. S., Chau, K.-W., & Emami, S.-M.-K. (2022). Prediction of future groundwater levels under representative concentration pathway scenarios using an inclusive multiple model coupled with artificial neural networks. *Journal of Water and Climate Change*, *13*(10), 3620-3643. https://doi.org/10.2166/wcc.2022.198
18. Galton, F., & British, A. (1885). Section H. Anthropology. Opening address by Francis Galton. *Nature*, *32*(830), 507-510.
19. Ghazi, B., Jeihouni, E., Kouzehgar, K., & Haghighi, A. T. (2021). Assessment of probable groundwater changes under representative concentration pathway (RCP) scenarios through the wavelet–GEP model. *Environmental Earth Sciences*, *80*(12), 446.
20. Gholami, V., Chau, K. W., Fadaee, F., Torkaman, J., & Ghaffari, A. (2015). Modeling of groundwater level fluctuations using dendrochronology in alluvial aquifers. *Journal of Hydrology*, *529*, 1060-1069.
21. Ghorbani, M. A., Deo, R. C., Yaseen, Z. M., H. Kashani, M., & Mohammadi, B. (2017). Pan evaporation prediction using a hybrid multilayer perceptron-firefly algorithm (MLP-FFA) model: case study in North Iran. *Theoretical and Applied Climatology*, *133*(3-4), 1119-1131. https://doi.org/10.1007/s00704-017-2244-0
22. Ghose, D., Das, U., & Roy, P. (2018). Modeling response of runoff and evapotranspiration for predicting water table depth in arid region using dynamic recurrent neural network. *Groundwater for Sustainable Development*, *6*, 263-269.
23. Guzman, S. M., Paz, J. O., & Tagert, M. L. M. (2017). The use of NARX neural networks to forecast daily groundwater levels. *Water Resources Management*, *31*, 1591-1603.
24. Guzman, S. M., Paz, J. O., Tagert, M. L. M., & Mercer, A. E. (2019). Evaluation of seasonally classified inputs for the prediction of daily groundwater levels: NARX networks vs support vector machines. *Environmental Modeling & Assessment*, *24*, 223-234.
25. Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks*, *5*(6), 989-993.
26. Hong, N., Hama, T., Suenaga, Y., Aqili, S. W., Huang, X., Wei, Q., & Kawagoshi, Y. (2016). Application of a modified conceptual rainfall–runoff model to simulation of groundwater level in an undefined watershed. *Science of The Total Environment*, *541*, 383-390.
27. Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, *2*(5), 359-366.
28. Huang, F., Huang, J., Jiang, S.-H., & Zhou, C. (2017). Prediction of groundwater levels using evidence of chaos and support vector machine. *Journal of Hydroinformatics*, *19*(4), 586-606.

29. Jalalkamali, A., & Jalalkamali, N. (2011). Groundwater modeling using hybrid of artificial neural network with genetic algorithm. *Afr. J. Agric. Res*, *6*(26), 5775-5784.
30. Jang, J.-S. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, *23*(3), 665-685.
31. Jha, M. K., & Sahoo, S. (2015). Efficacy of neural network and genetic algorithm techniques in simulating spatio-temporal fluctuations of groundwater. *Hydrological processes*, *29*(5), 671-691.
32. Kashiwao, T., Nakayama, K., Ando, S., Ikeda, K., Lee, M., & Bahadori, A. (2017). A neural network-based local rainfall prediction system using meteorological data on the Internet: A case study using data from the Japan Meteorological Agency. *Applied Soft Computing*, *56*, 317-330. https://doi.org/10.1016/j.asoc.2017.03.015
33. Kazemi, H., Sarukkalige, R., & Shao, Q. (2021). Evaluation of non-uniform groundwater level data using spatiotemporal modeling. *Groundwater for Sustainable Development*, *15*, 100659.
34. Kennedy, J. (1997). The particle swarm: social adaptation of knowledge. Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC'97),
35. Khorasani, M., Ehteshami, M., Ghadimi, H., & Salari, M. (2016). Simulation and analysis of temporal changes of groundwater depth using time series modeling. *Modeling Earth Systems and Environment*, *2*(2). https://doi.org/10.1007/s40808-016-0164-0
36. Kişi, Ö. (2010). River suspended sediment concentration modeling using a neural differential evolution approach. *Journal of Hydrology*, *389*(1-2), 227-235.
37. Kisi, O., & Shiri, J. (2012). River suspended sediment estimation by climatic variables implication: comparative study among soft computing techniques. *Computers & Geosciences*, *43*, 73-82.
38. Kombo, O., Kumaran, S., Sheikh, Y., Bovim, A., & Jayavel, K. (2020). Long-Term Groundwater Level Prediction Model Based on Hybrid KNN-RF Technique. *Hydrology*, *7*(3). https://doi.org/10.3390/hydrology7030059
39. Lallahem, S., Mania, J., Hani, A., & Najjar, Y. (2005). On the use of neural networks to evaluate groundwater levels in fractured media. *Journal of Hydrology*, *307*(1-4), 92-111. https://doi.org/10.1016/j.jhydrol.2004.10.005
40. Lee, S., Lee, K.-K., & Yoon, H. (2019). Using artificial neural network models for groundwater level forecasting and assessment of the relative impacts of influencing factors. *Hydrogeology Journal*, *27*(2).
41. Li, F., Zhao, Y., Feng, P., Zhang, W., & Qiao, J. (2015). Risk assessment of groundwater and its application. Part I: risk grading based on the functional zoning of groundwater. *Water Resources Management*, *29*, 2697-2714.
42. Liang, X., & Zhang, Y.-K. (2015). Analyses of uncertainties and scaling<? xmltex\newline?> of groundwater level fluctuations. *Hydrology and Earth System Sciences*, *19*(7), 2971-2979.
43. Lyazidi, R., Hessane, M. A., Moutei, J. F., & Bahir, M. (2020). Developing a methodology for estimating the groundwater levels of coastal aquifers in the Gareb-Bourag plains, Morocco embedding the visual

MODFLOW techniques in groundwater modeling system. *Groundwater for Sustainable Development*, *11*, 100471.

44. Maiti, S., & Tiwari, R. K. (2013). A comparative study of artificial neural networks, Bayesian neural networks and adaptive neuro-fuzzy inference system in groundwater level prediction. *Environmental Earth Sciences*, *71*(7), 3147-3160. https://doi.org/10.1007/s12665-013-2702-7
45. Malekinezhad, H., & Banadkooki, F. B. (2018). Modeling impacts of climate change and human activities on groundwater resources using MODFLOW. *Journal of Water and Climate Change*, *9*(1), 156-177 %@ 2040-2244.
46. Malik, A., & Bhagwat, A. (2021). Modelling groundwater level fluctuations in urban areas using artificial neural network. *Groundwater for Sustainable Development*, *12*, 100484.
47. Moghaddam, H. K., Moghaddam, H. K., Kivi, Z. R., Bahreinimotlagh, M., & Alizadeh, M. J. (2019). Developing comparative mathematic models, BN and ANN for forecasting of groundwater levels. *Groundwater for Sustainable Development*, *9*, 100237.
48. Mohanty, S., Jha, M. K., Raul, S., Panda, R., & Sudheer, K. (2015). Using artificial neural network approach for simultaneous forecasting of weekly groundwater levels at multiple sites. *Water Resources Management*, *29*, 5521-5532.
49. Nadiri, A. A., Naderi, K., Khatibi, R., & Gharekhani, M. (2019). Modelling groundwater level variations by learning from multiple models using fuzzy logic. *Hydrological Sciences Journal*, *64*(2), 210-226. https://doi.org/10.1080/02626667.2018.1554940
50. Natarajan, N., & Sudheer, C. (2019). Groundwater level forecasting using soft computing techniques. *Neural Computing and Applications*, *32*(12), 7691-7708. https://doi.org/10.1007/s00521-019-04234-5
51. Noori, R., Maghrebi, M., Mirchi, A., Tang, Q., Bhattarai, R., Sadegh, M., Noury, M., Torabi Haghighi, A., Kløve, B., & Madani, K. (2021). Anthropogenic depletion of Iran's aquifers. *Proceedings of the National Academy of Sciences*, *118*(25), e2024221118.
52. Nourani, V., Alami, M. T., & Vousoughi, F. D. (2015). Wavelet-entropy data pre-processing approach for ANN-based groundwater level modeling. *Journal of Hydrology*, *524*, 255-269.
53. Nourani, V., Hosseini Baghanam, A., Adamowski, J., & Kisi, O. (2014). Applications of hybrid wavelet–Artificial Intelligence models in hydrology: A review. *Journal of Hydrology*, *514*, 358-377. https://doi.org/10.1016/j.jhydrol.2014.03.057
54. Nourani, V., & Mousavi, S. (2016). Spatiotemporal groundwater level modeling using hybrid artificial intelligence-meshless method. *Journal of Hydrology*, *536*, 10-25. https://doi.org/10.1016/j.jhydrol.2016.02.030
55. Pandey, K., Kumar, S., Malik, A., & Kuriqi, A. (2020). Artificial Neural Network Optimized with a Genetic Algorithm for Seasonal Groundwater Table Depth Prediction in Uttar Pradesh, India. *Sustainability*, *12*(21). https://doi.org/10.3390/su12218932
56. Park, Y.-S., & Lek, S. (2016). Artificial neural networks: multilayer perceptron for ecological modeling. In *Developments in environmental modelling* (Vol. 28, pp. 123-140). Elsevier.

57. Peng, T., Zhou, J., Zhang, C., & Fu, W. (2017). Streamflow forecasting using empirical wavelet transform and artificial neural networks. *Water*, *9*(6), 406.
58. Pham, Q. B., Abba, S. I., Usman, A. G., Linh, N. T. T., Gupta, V., Malik, A., Costache, R., Vo, N. D., & Tri, D. Q. (2019). Potential of Hybrid Data-Intelligence Algorithms for Multi-Station Modelling of Rainfall. *Water Resources Management*, *33*(15), 5067-5087. https://doi.org/10.1007/s11269-019-02408-3
59. Rajaee, T., Ebrahimi, H., & Nourani, V. (2019). A review of the artificial intelligence methods in groundwater level modeling. *Journal of Hydrology*, *572*, 336-351.
60. Rajaee, T., Mirbagheri, S. A., Zounemat-Kermani, M., & Nourani, V. (2009). Daily suspended sediment concentration simulation using ANN and neuro-fuzzy models. *Science of The Total Environment*, *407*(17), 4916-4927.
61. Rakhshandehroo, G. R., Vaghefi, M., & Aghbolaghi, M. A. (2012). Forecasting Groundwater Level in Shiraz Plain Using Artificial Neural Networks. *Arabian Journal for Science and Engineering*, *37*(7), 1871-1883. https://doi.org/10.1007/s13369-012-0291-5
62. Ranjithan, S. (1993). Neural Network-Based Screening for Groundwater Reclamation Under Uncertainty. *Water Resources Research*, *29*(3), 11.
63. Retike, I., Bikše, J., Kalvāns, A., Dēliņa, A., Avotniece, Z., Zaadnoordijk, W. J., Jemeljanova, M., Popovs, K., Babre, A., Zelenkevičs, A., & Baikovs, A. (2022). Rescue of groundwater level time series: How to visually identify and treat errors. *Journal of Hydrology*, *605*. https://doi.org/10.1016/j.jhydrol.2021.127294
64. Rogers, L. L. Optimization of groundwater remediation using artificial neural networks with parallel solute transport modeling. *Water Resources Research*, *30*(2), 24.
65. Roshni, T., Jha, M. K., & Drisya, J. (2020). Neural network modeling for groundwater-level forecasting in coastal aquifers. *Neural Computing and Applications*, *32*, 12737-12754.
66. Sahoo, S., Russo, T., Elliott, J., & Foster, I. (2017). Machine learning algorithms for modeling groundwater level changes in agricultural regions of the US. *Water Resources Research*, *53*(5), 3878-3895.
67. Sheng, M., Liu, J., Zhu, A.-X., Rossiter, D. G., Liu, H., Liu, Z., & Zhu, L. (2019). Comparison of GLUE and DREAM for the estimation of cultivar parameters in the APSIM-maize model. *Agricultural and Forest Meteorology*, *278*, 107659.
68. Sudheer, C., & Shashi, M. (2012). Groundwater level forecasting using SVM-PSO. *International Journal of Hydrology Science and Technology*, *2*(2), 202-218.
69. Sun, Y., Wendi, D., Kim, D. E., & Liong, S.-Y. (2016). Application of artificial neural networks<? xmltex\break?> in groundwater table forecasting–a case study in a<? xmltex\break?> Singapore swamp forest. *Hydrology and Earth System Sciences*, *20*(4), 1405-1412.
70. Supreetha, B., Nayak, K. P., & Shenoy, K. N. (2019). Hybrid artificial intelligence based abc-pso system for ground water level forecasting in udupi region. *Journal of Engineering Science and Technology*, *14*(2), 797-809.

71. Tahmasebi, P., & Hezarkhani, A. (2009). Application of optimized neural network by genetic algorithm, IAMG09. In: Stanford University, California.
72. Tang, Z., De Almeida, C., & Fishwick, P. A. (1991). Time series forecasting using neural networks vs. Box-Jenkins methodology. *Simulation*, *57*(5), 303-310.
73. Torabi Haghighi, A., Abou Zaki, N., Rossi, P. M., Noori, R., Hekmatzadeh, A. A., Saremi, H., & Kløve, B. (2020). Unsustainability syndrome—from meteorological to agricultural drought in arid and semi-arid regions. *Water*, *12*(3), 838.
74. Trưởng, N. H., & Dao, D.-N. (2020). New hybrid between NSGA-III with multi-objective particle swarm optimization to multi-objective robust optimization design for Powertrain mount system of electric vehicles. *Advances in Mechanical Engineering*, *12*(2), 1687814020904253 %@ 1687814020901687-1687814020908140.
75. Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, *30*(1), 79-82 %@ 0936-0577X.
76. Yadav, A., Chatterjee, S., & Equeenuddin, S. M. (2018). Prediction of suspended sediment yield by artificial neural network and traditional mathematical model in Mahanadi river basin, India. *Sustainable Water Resources Management*, *4*, 745-759.
77. Yadav, A., Chatterjee, S., & Equeenuddin, S. M. (2021). Suspended sediment yield modeling in Mahanadi River, India by multi-objective optimization hybridizing artificial intelligence algorithms. *International Journal of Sediment Research*, *36*(1), 76-91 %@ 1001-6279.
78. Yang, T.-h., Shi, W.-h., Wang, P.-t., Liu, H.-l., Yu, Q.-l., & Li, Y. (2015). Numerical simulation on slope stability analysis considering anisotropic properties of layered fractured rocks: a case study. *Arabian Journal of Geosciences*, *8*, 5413-5421.
79. Yaseen, Z., Ebtehaj, I., Kim, S., Sanikhani, H., Asadi, H., Ghareb, M., Bonakdari, H., Wan Mohtar, W., Al-Ansari, N., & Shahid, S. (2019). Novel Hybrid Data-Intelligence Model for Forecasting Monthly Rainfall with Uncertainty Analysis. *Water*, *11*(3). https://doi.org/10.3390/w11030502
80. Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing 50*(2003), 16.
81. Zhang, J., Zhang, X., Niu, J., Hu, B. X., Soltanian, M. R., Qiu, H., & Yang, L. (2019). Prediction of groundwater level in seashore reclaimed land using wavelet and artificial neural network-based hybrid model. *Journal of Hydrology*, *577*, 123948.
82. Zhang, R., Chen, S., Zhang, Z., & Zhu, W. (2022). Genetic Algorithm in Multimedia Dynamic Prediction of Groundwater in Open-Pit Mine. *Comput Intell Neurosci*, *2022*, 8556103. https://doi.org/10.1155/2022/8556103
83. Zhou, T., Wang, F., & Yang, Z. (2017). Comparative analysis of ANN and SVM models combined with wavelet preprocess for groundwater depth prediction. *Water*, *9*(10), 781.

Figures

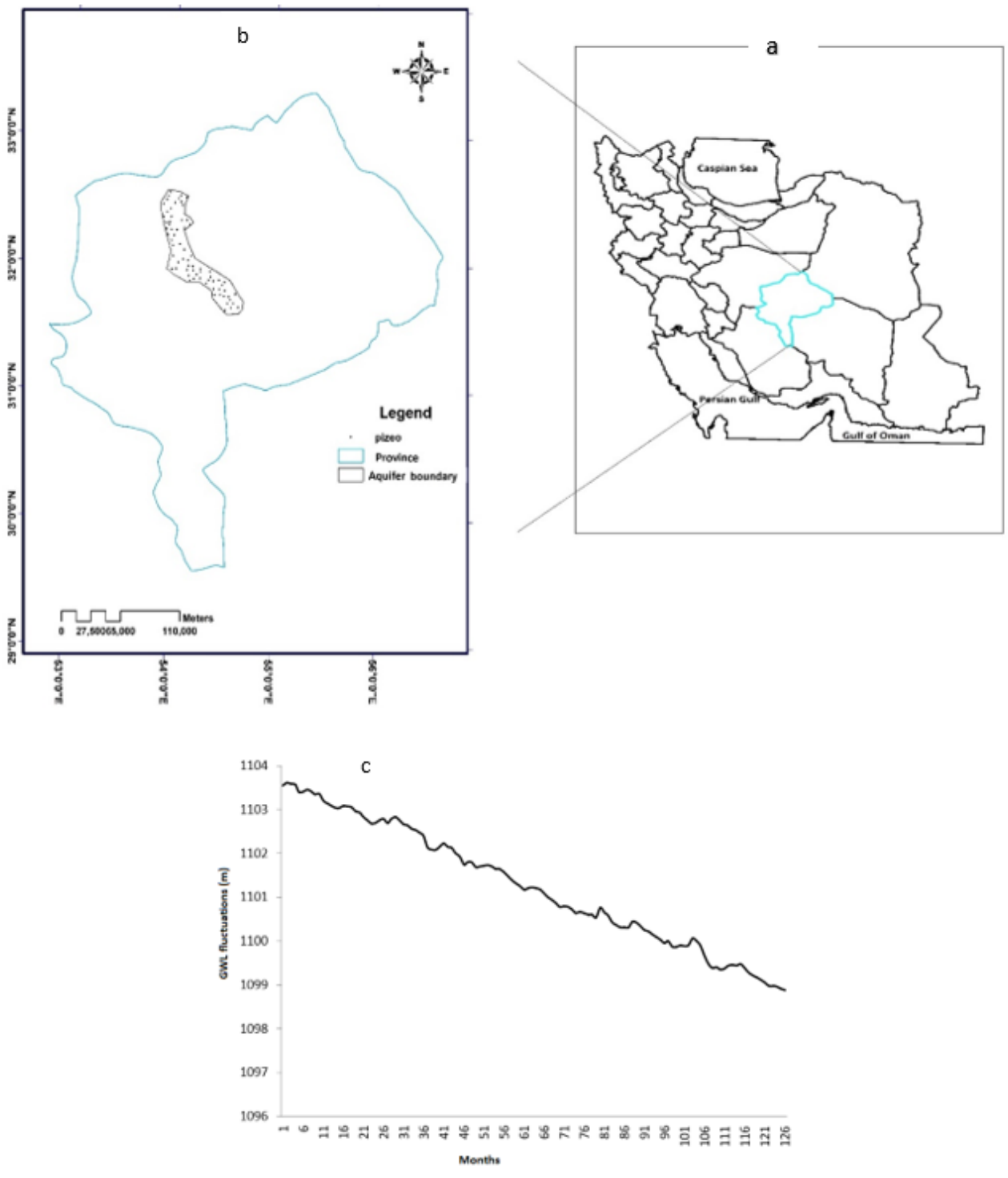

Figure 1

Study area in Iran (a).Location of piezometer across Yazd-Ardakan aquifer (b) the and water level fluctuation (c).

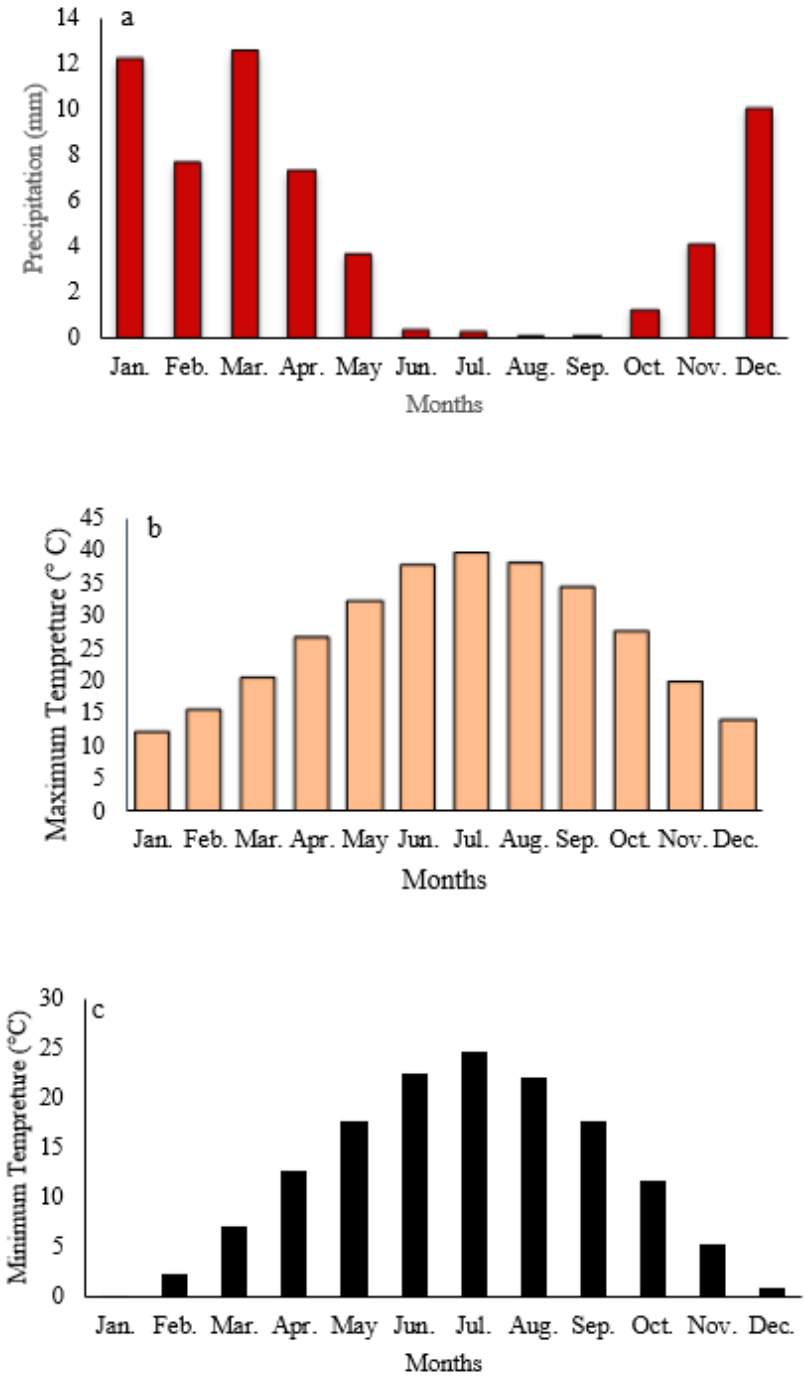

Figure 2

Mean monthly precipitation (a), maximum temperature (b), and minimum temperature (c) of the Yazd-Ardakan aquifer (based on the Yazd station)

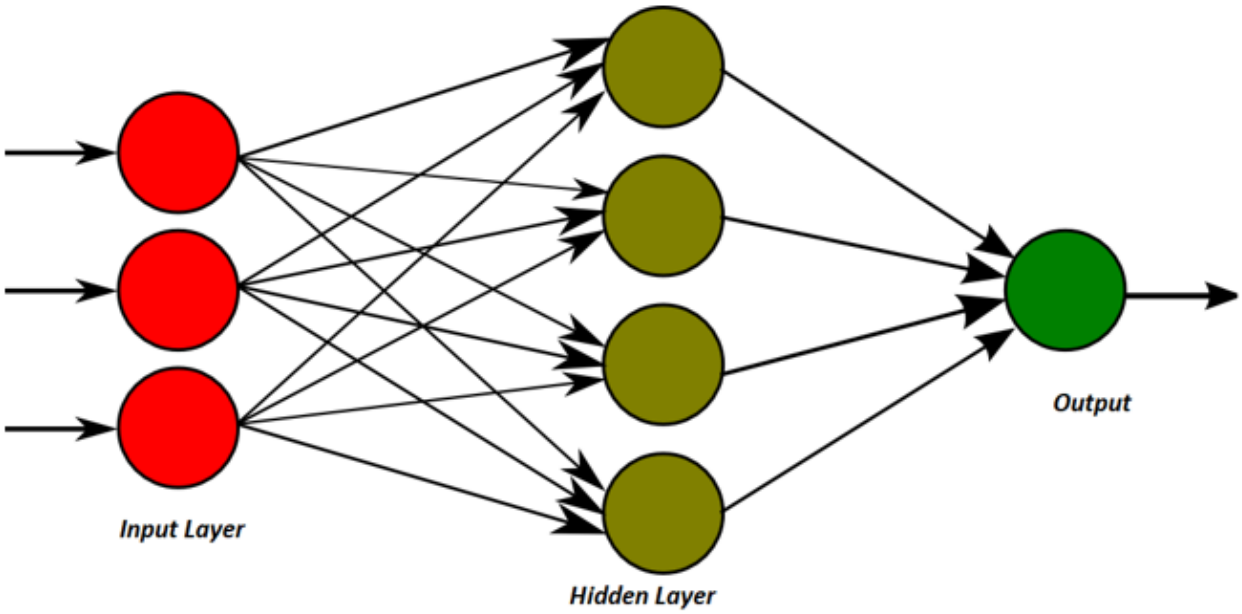

Figure 3

Multilayer Perceptron network structure

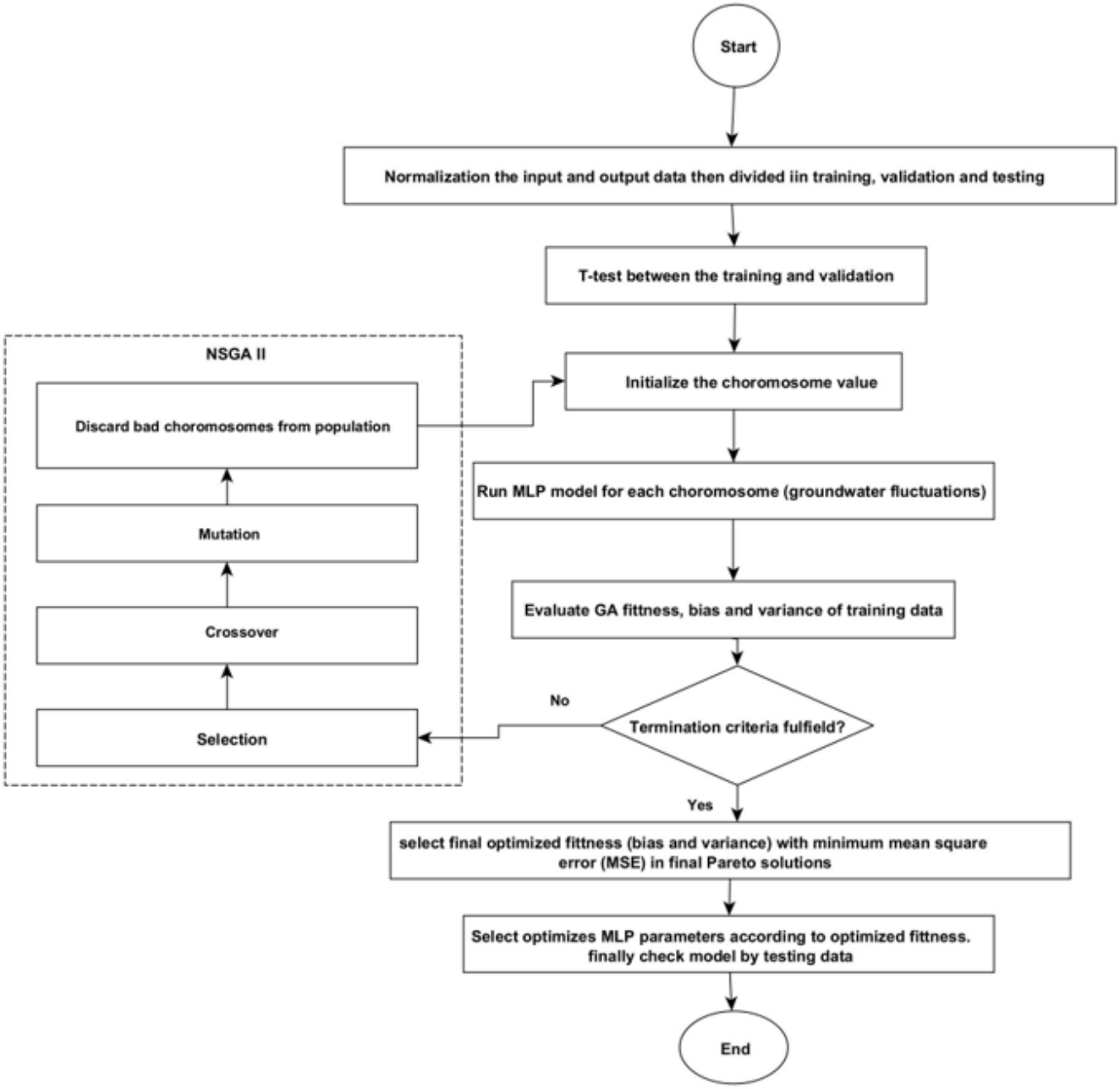

Figure 4

MLP-NSGA-II schematic diagram

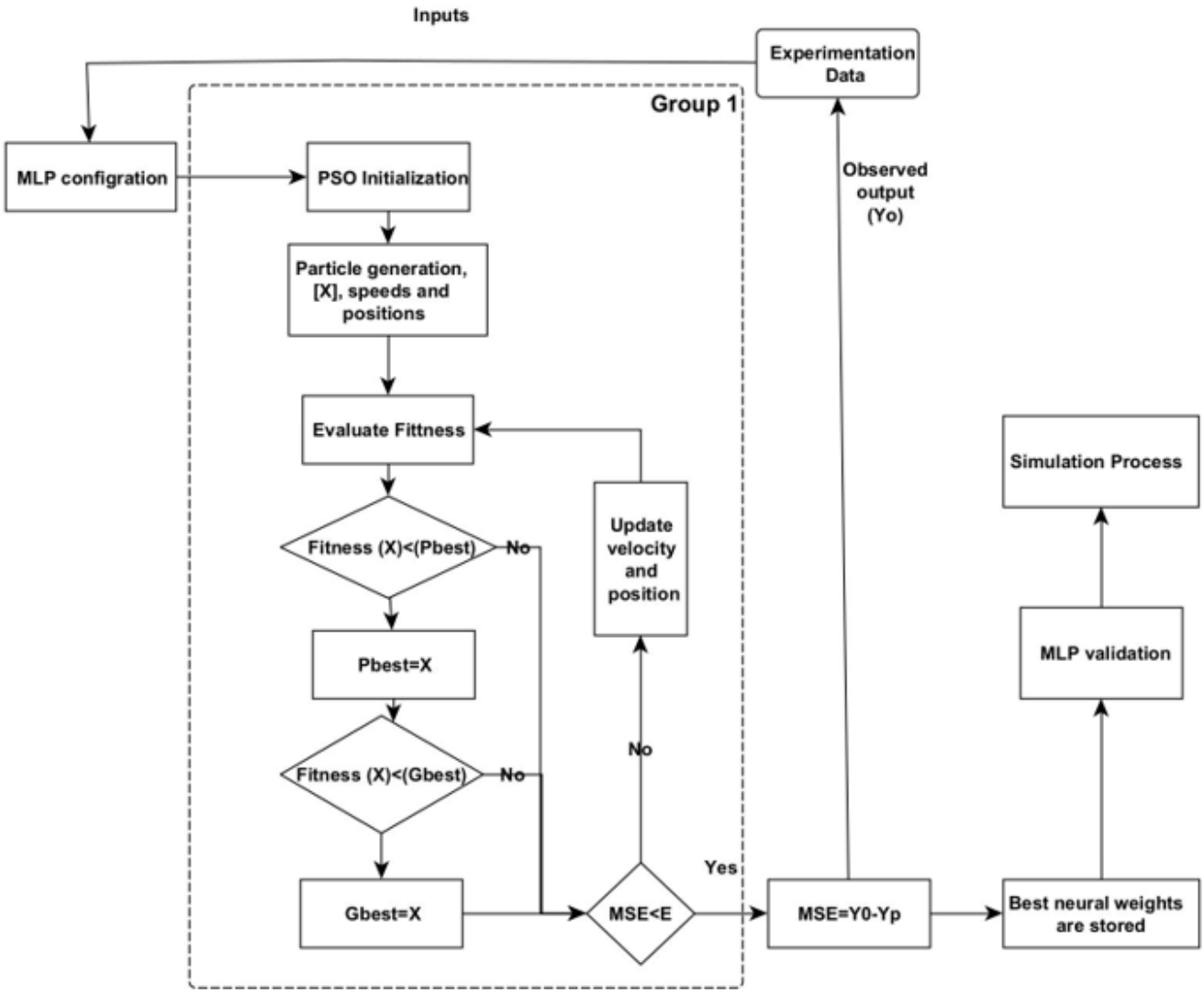

Figure 5

The flowchart of MOPSO-MLP

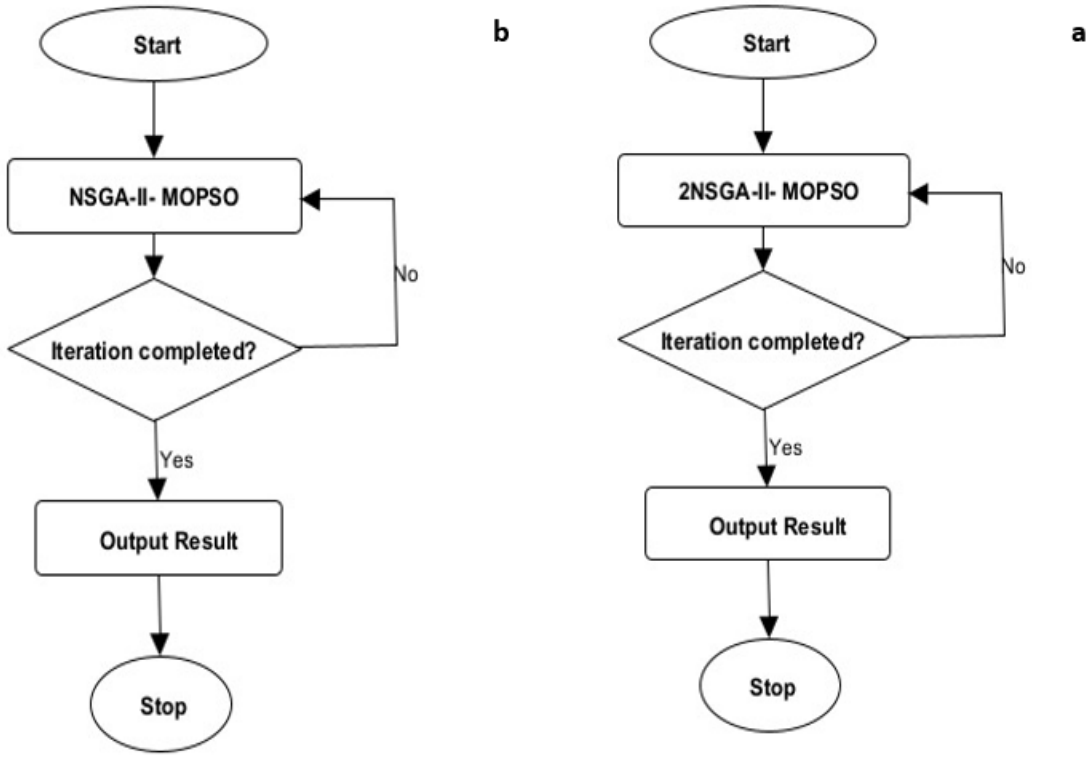

Figure 6

(a) Hybrid model of MLP-2NSGA-II-MOPSO- and (b) MLP-MOPSO-NSGA-II.

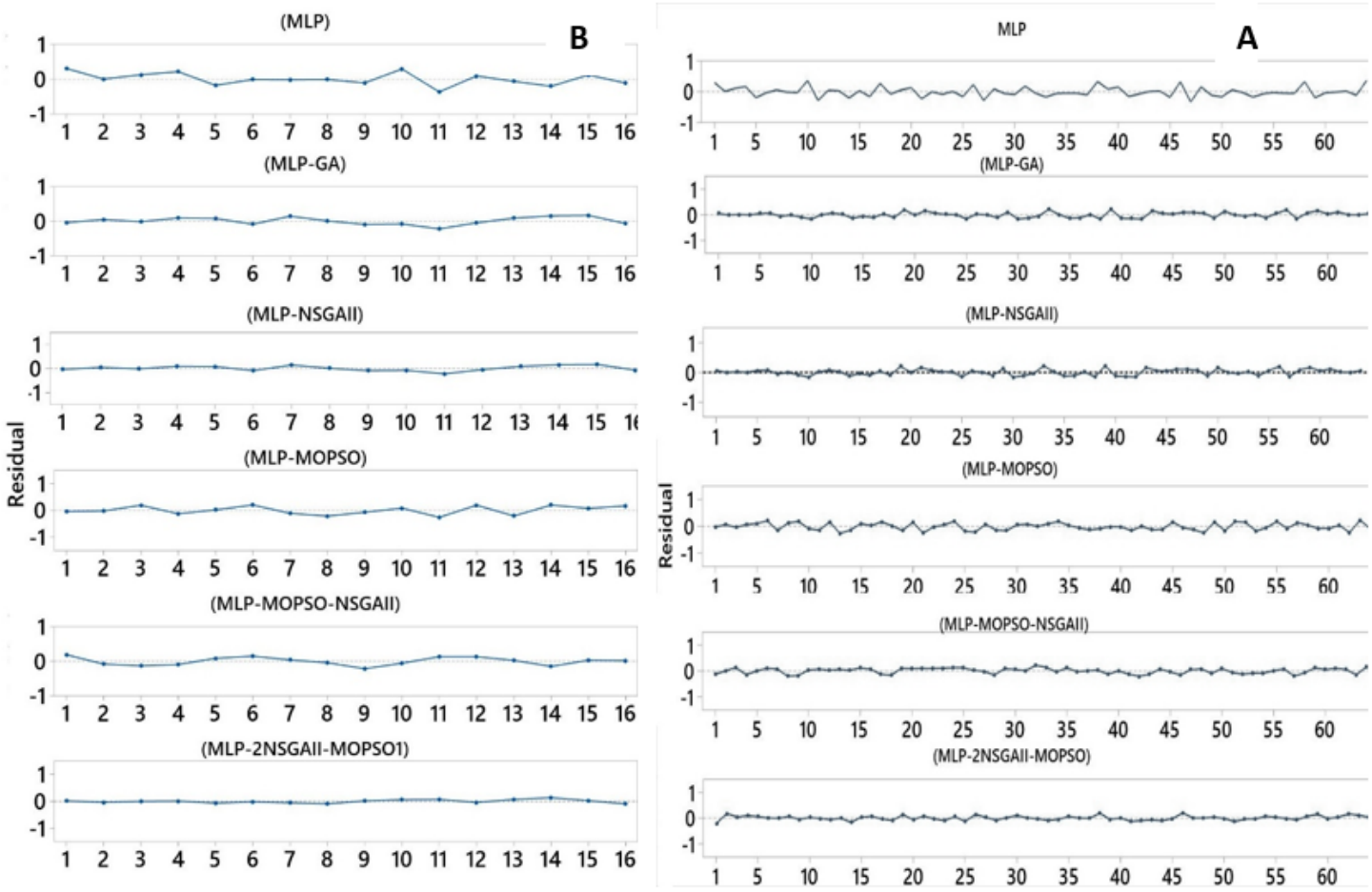

Figure 7

Residual plots of different models in the training (A) and testing (B) stages.

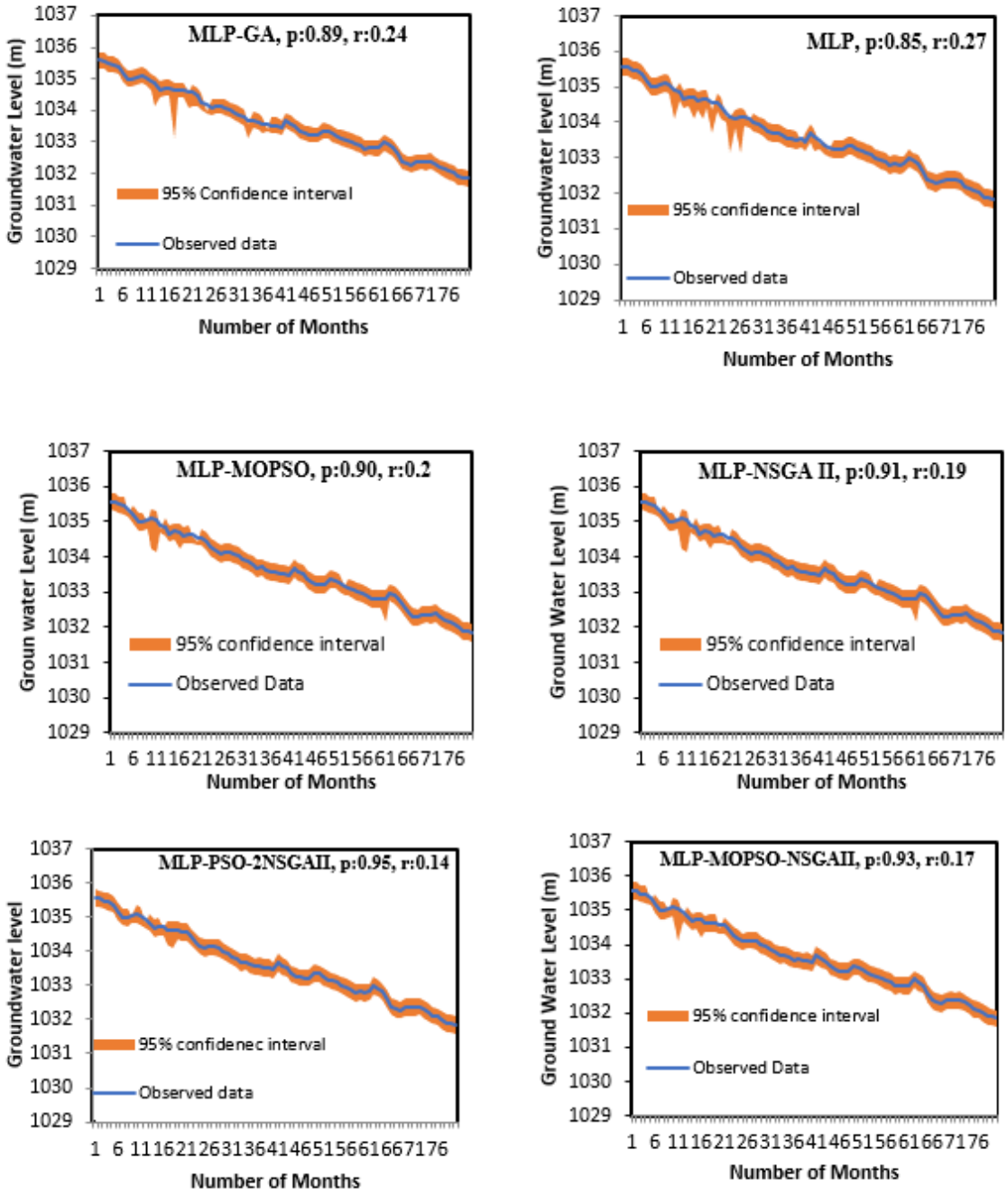

Figure 8

The uncertainty values of the models